FIG. 1

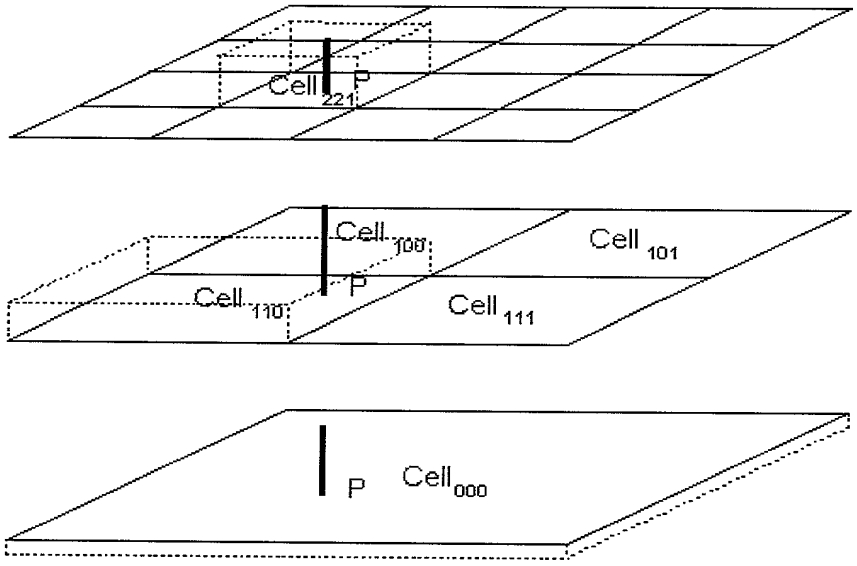


FIG. 2

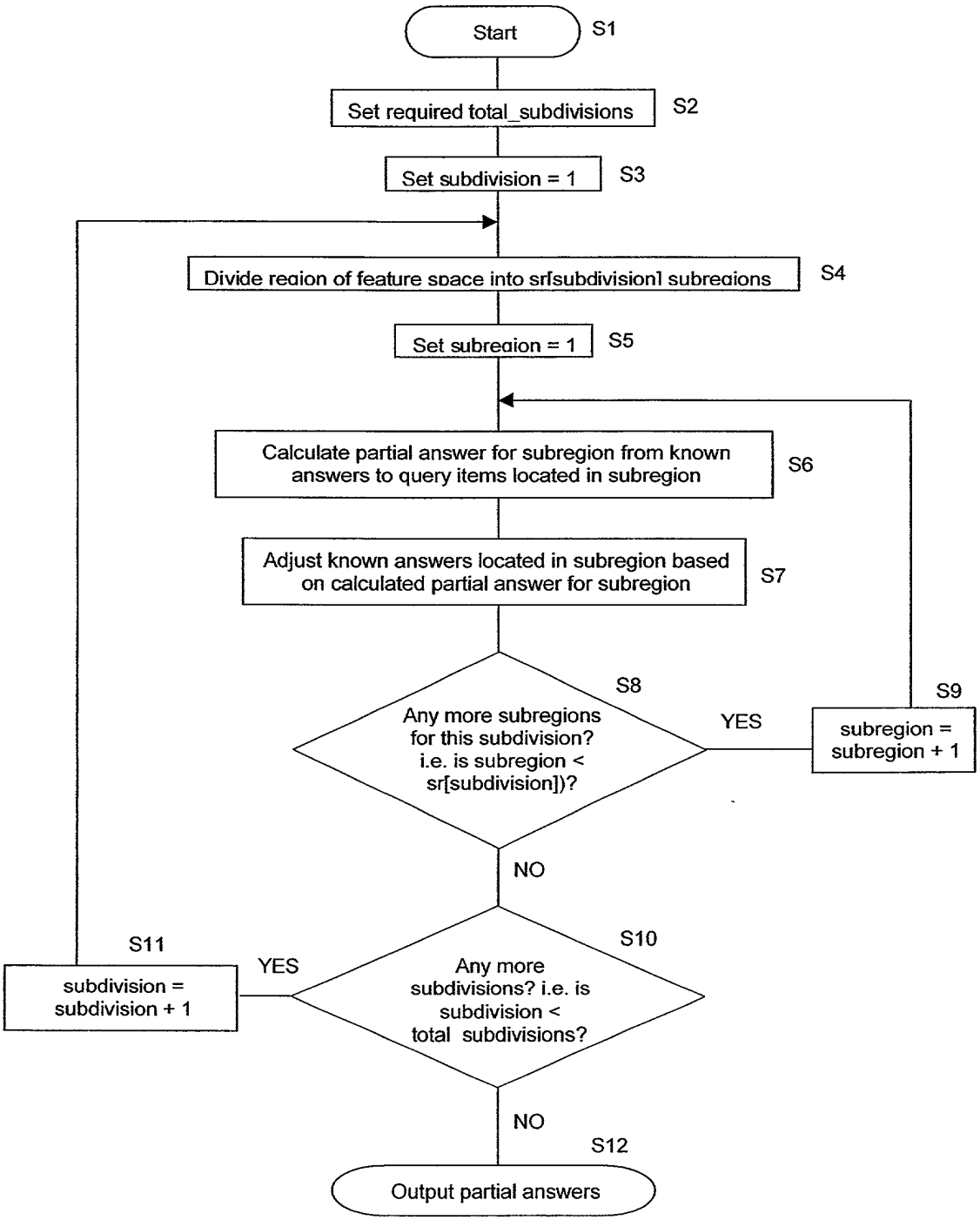

Start
S1
Set required total_subdivisions
S2
Set subdivision = 1
S3
Divide region of feature space into sr[subdivision] subregions
S4
Set subregion = 1
S5
Calculate partial answer for subregion from known answers to query items located in subregion
S6
Adjust known answers located in subregion based on calculated partial answer for subregion
S7
S8
Any more subregions for this subdivision? i.e. is subregion < sr[subdivision])?
YES
S9
subregion = subregion + 1
NO
S10
Any more subdivisions? i.e. is subdivision < total subdivisions?
YES
S11
subdivision = subdivision + 1
NO
S12
Output partial answers

FIG. 3

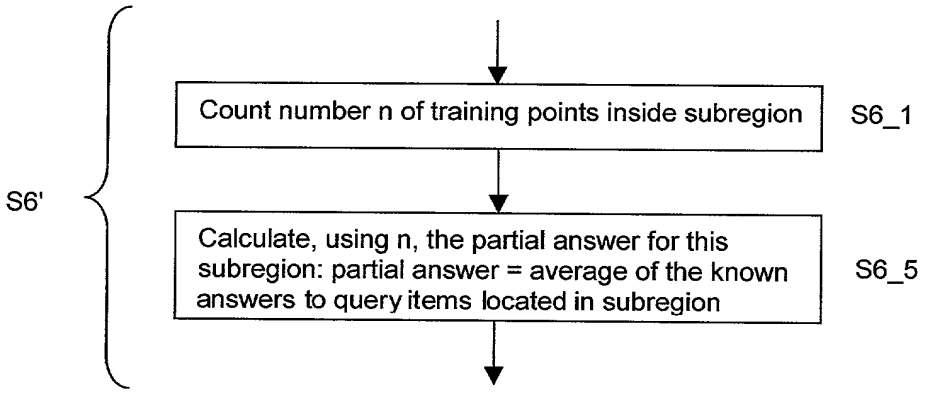

S6'
Count number n of training points inside subregion
S6_1
Calculate, using n, the partial answer for this subregion: partial answer = average of the known answers to query items located in subregion
S6_5

FIG. 4

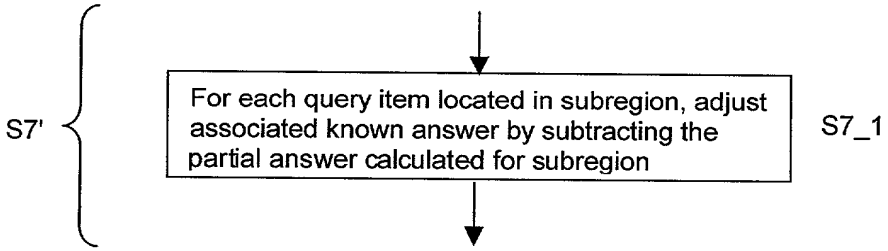

S7'
For each query item located in subregion, adjust associated known answer by subtracting the partial answer calculated for subregion
S7_1

FIG. 5

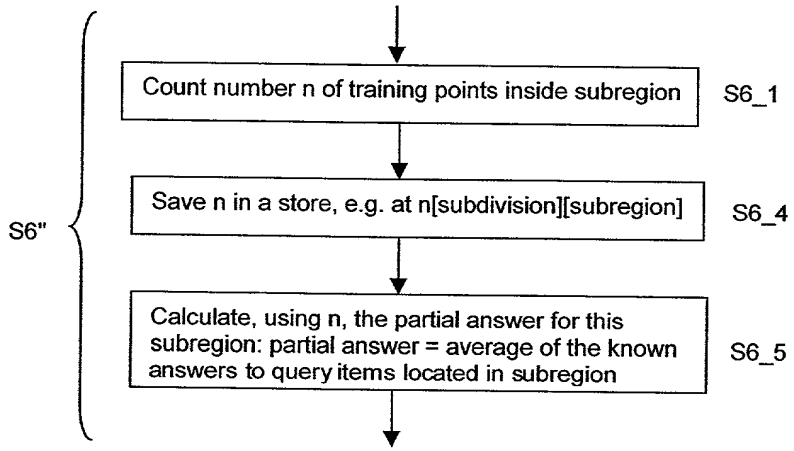

S6"
Count number n of training points inside subregion
S6_1
Save n in a store, e.g. at n[subdivision][subregion]
S6_4
Calculate, using n, the partial answer for this subregion: partial answer = average of the known answers to query items located in subregion
S6_5

FIG. 6

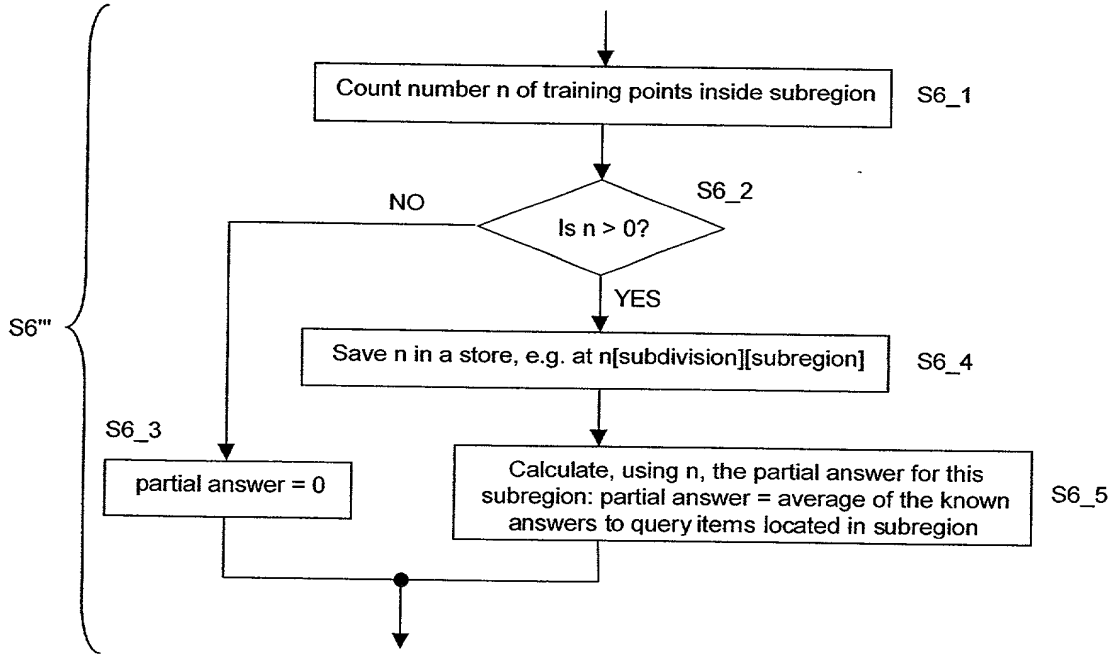

S6'''
Count number n of training points inside subregion
S6_1
S6_2
Is n > 0?
NO
YES
Save n in a store, e.g. at n[subdivision][subregion]
S6_4
S6_3
partial answer = 0
Calculate, using n, the partial answer for this subregion: partial answer = average of the known answers to query items located in subregion
S6_5

FIG. 7

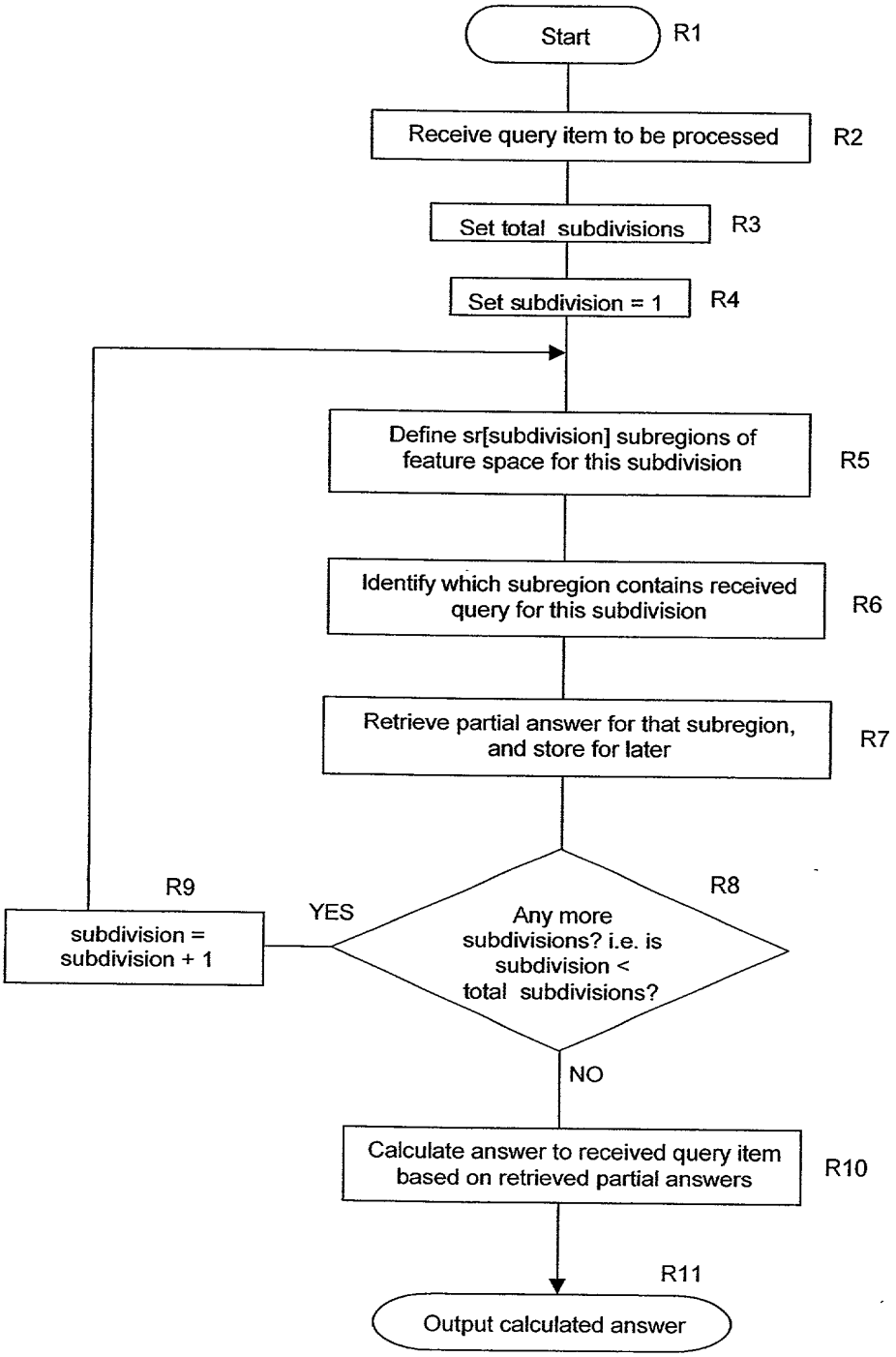

Start
R1
Receive query item to be processed
R2
Set total subdivisions
R3
Set subdivision = 1
R4
Define sr[subdivision] subregions of feature space for this subdivision
R5
Identify which subregion contains received query for this subdivision
R6
Retrieve partial answer for that subregion, and store for later
R7
R8
Any more subdivisions? i.e. is subdivision < total subdivisions?
YES
R9
subdivision = subdivision + 1
NO
Calculate answer to received query item based on retrieved partial answers
R10
R11
Output calculated answer

FIG. 8

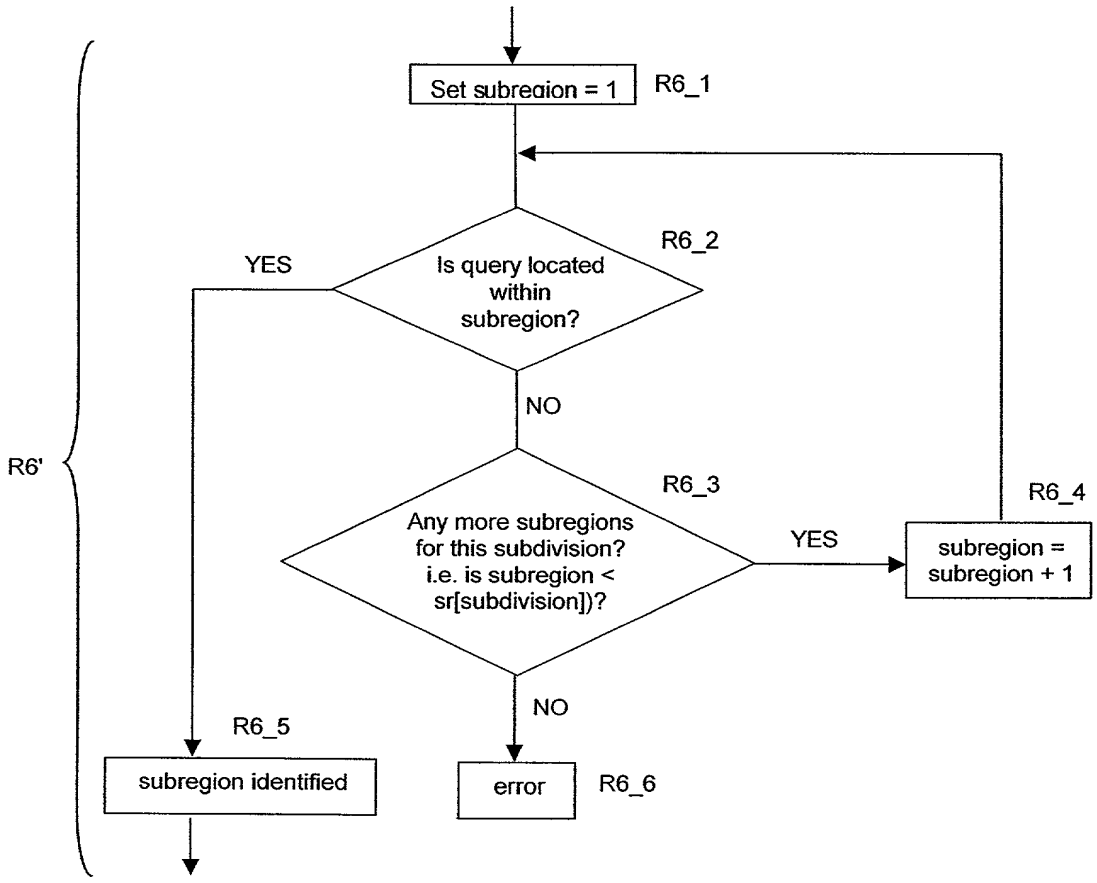

R6'
Set subregion = 1
R6_1
R6_2
Is query located within subregion?
YES
NO
R6_3
Any more subregions for this subdivision? i.e. is subregion < sr[subdivision])?
YES
R6_4
subregion = subregion + 1
NO
R6_5
subregion identified
error
R6_6

FIG. 9

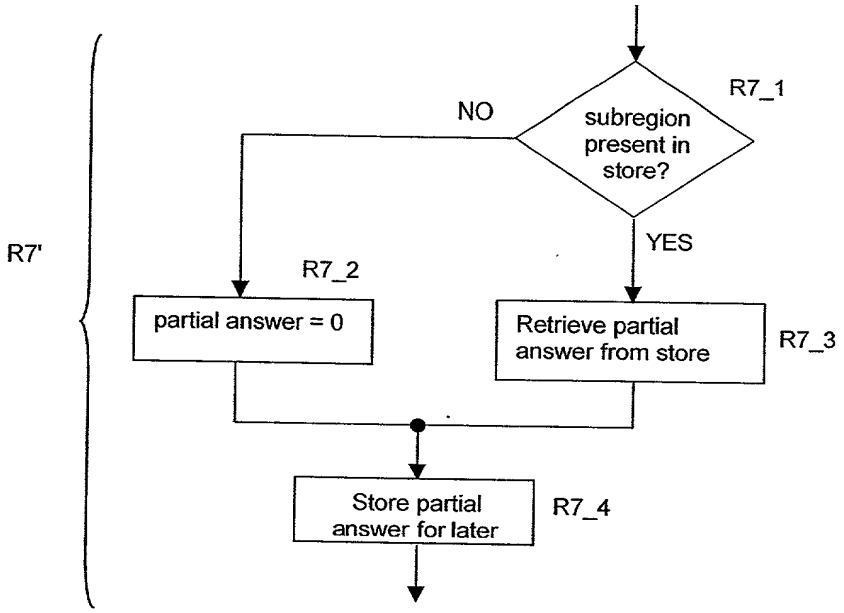

R7'
R7_1
subregion present in store?
NO
YES
R7_2
partial answer = 0
Retrieve partial answer from store
R7_3
Store partial answer for later
R7_4

FIG. 10

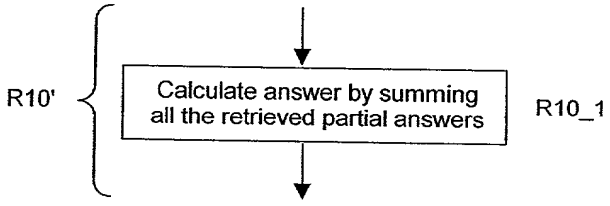

R10'
Calculate answer by summing all the retrieved partial answers
R10_1

FIG. 11

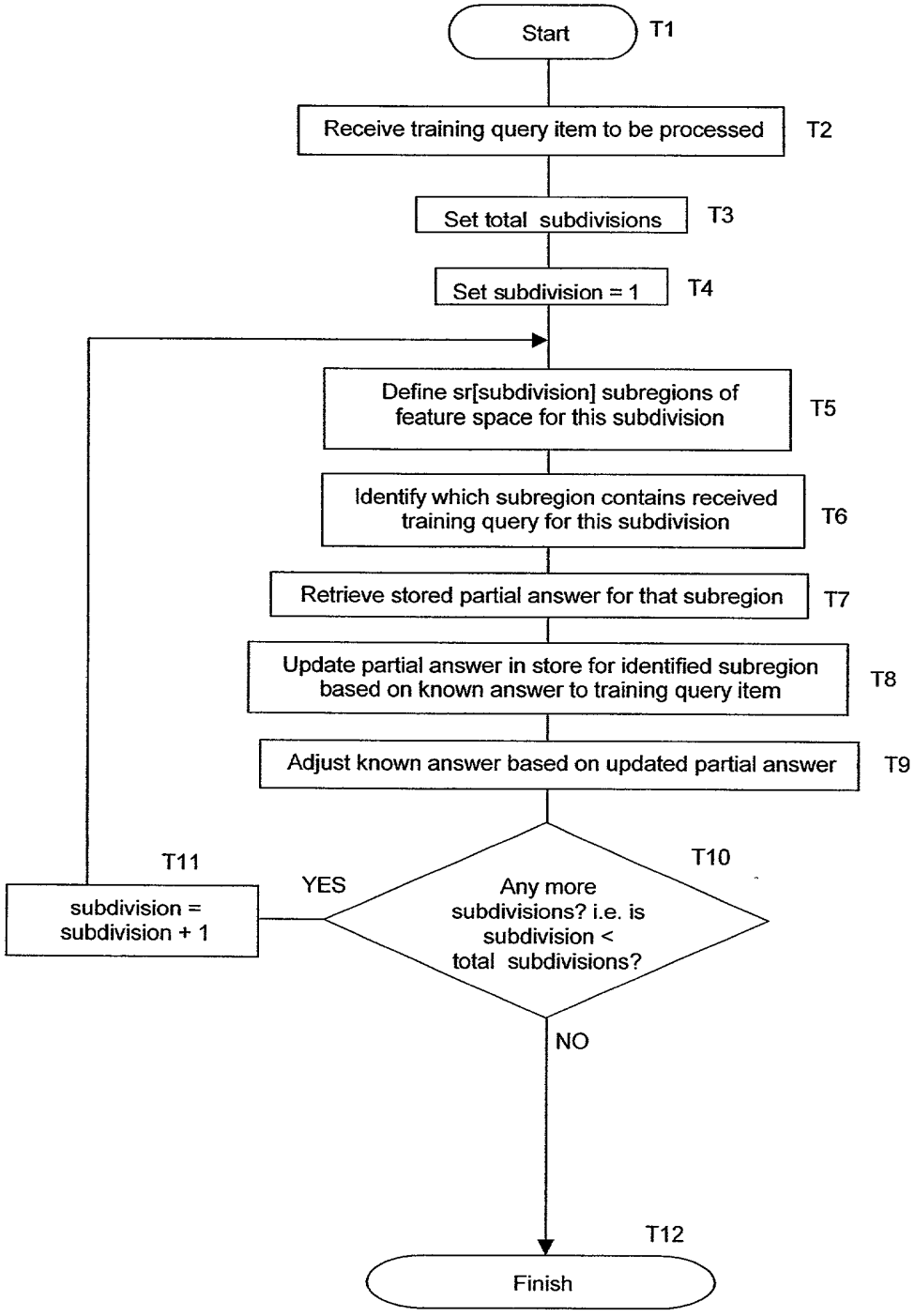

Start
T1
Receive training query item to be processed
T2
Set total subdivisions
T3
Set subdivision = 1
T4
Define sr[subdivision] subregions of feature space for this subdivision
T5
Identify which subregion contains received training query for this subdivision
T6
Retrieve stored partial answer for that subregion
T7
Update partial answer in store for identified subregion based on known answer to training query item
T8
Adjust known answer based on updated partial answer
T9
Any more subdivisions? i.e. is subdivision < total subdivisions?
T10
YES
subdivision = subdivision + 1
T11
NO
Finish
T12

FIG. 12

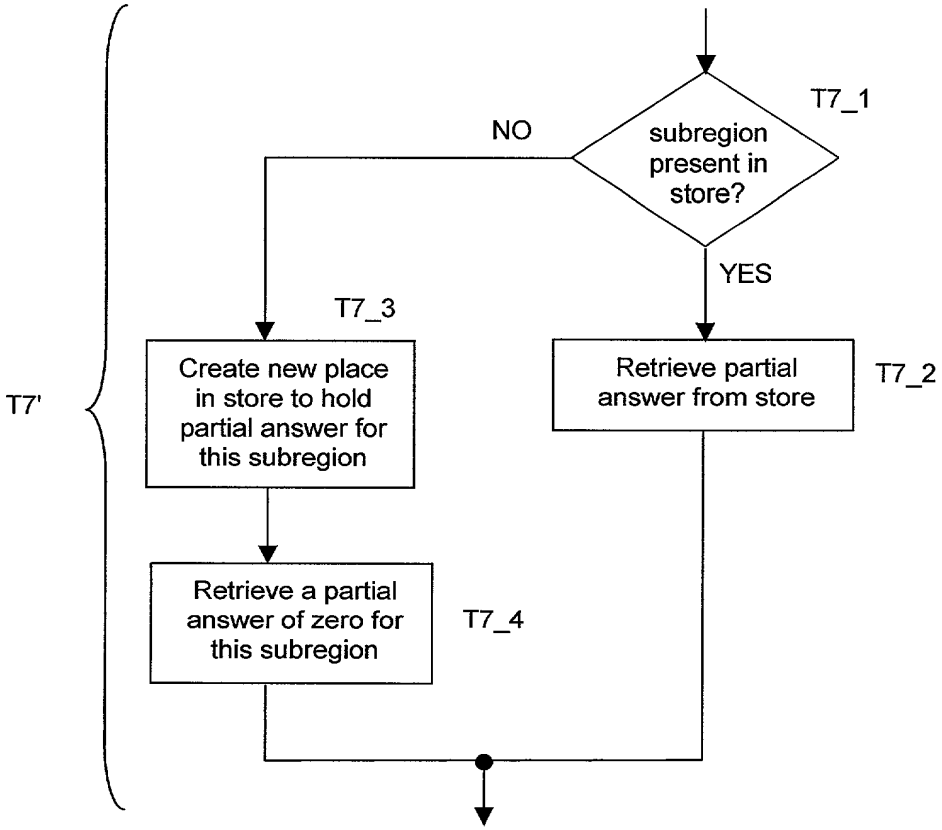

T7'
T7_1
subregion present in store?
NO
YES
T7_3
Create new place in store to hold partial answer for this subregion
Retrieve partial answer from store
T7_2
Retrieve a partial answer of zero for this subregion
T7_4

FIG. 13

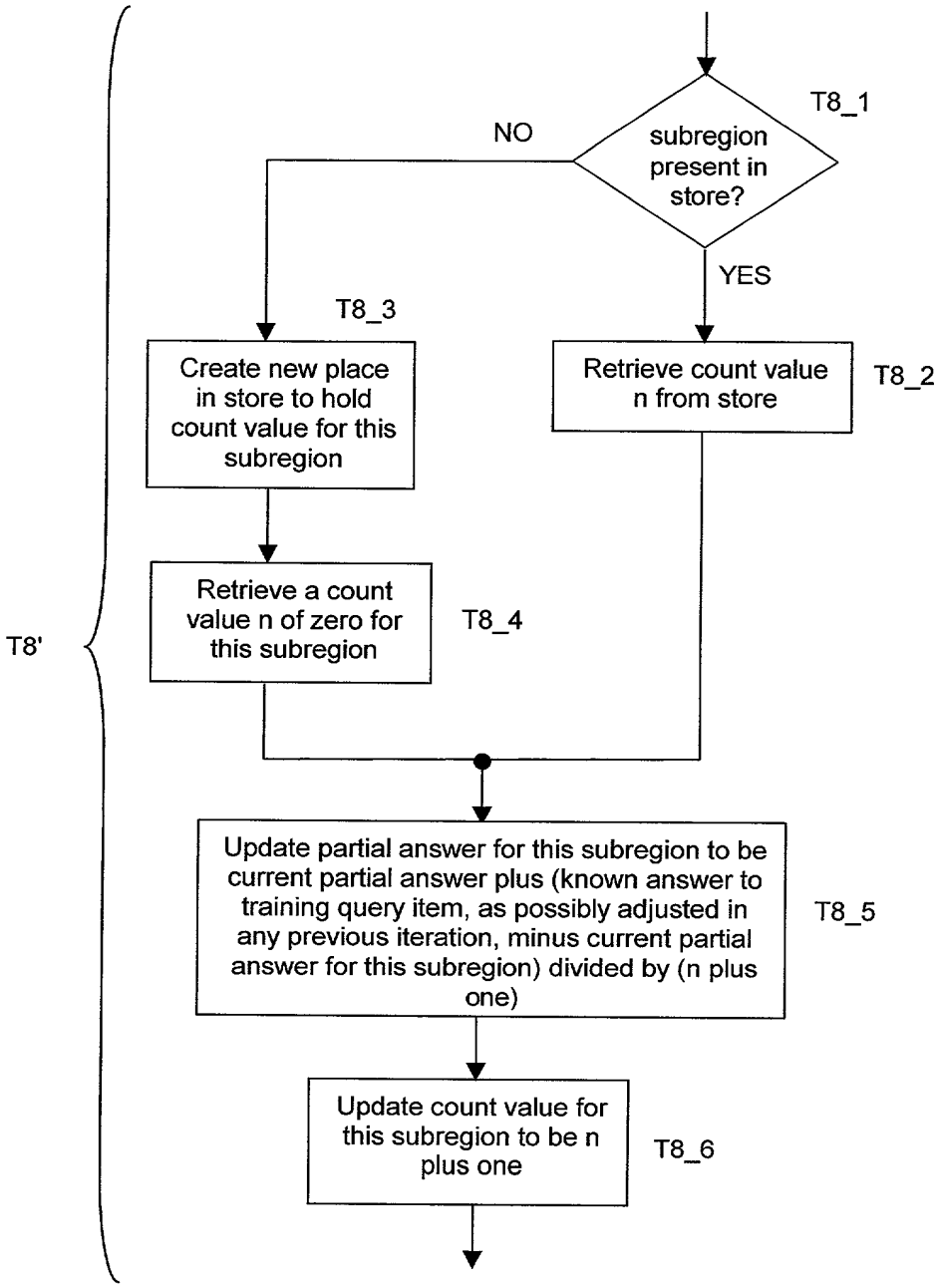

T8'
T8_1
subregion present in store?
NO
YES
T8_3
Create new place in store to hold count value for this subregion
Retrieve count value n from store
T8_2
Retrieve a count value n of zero for this subregion
T8_4
Update partial answer for this subregion to be current partial answer plus (known answer to training query item, as possibly adjusted in any previous iteration, minus current partial answer for this subregion) divided by (n plus one)
T8_5
Update count value for this subregion to be n plus one
T8_6

FIG. 14

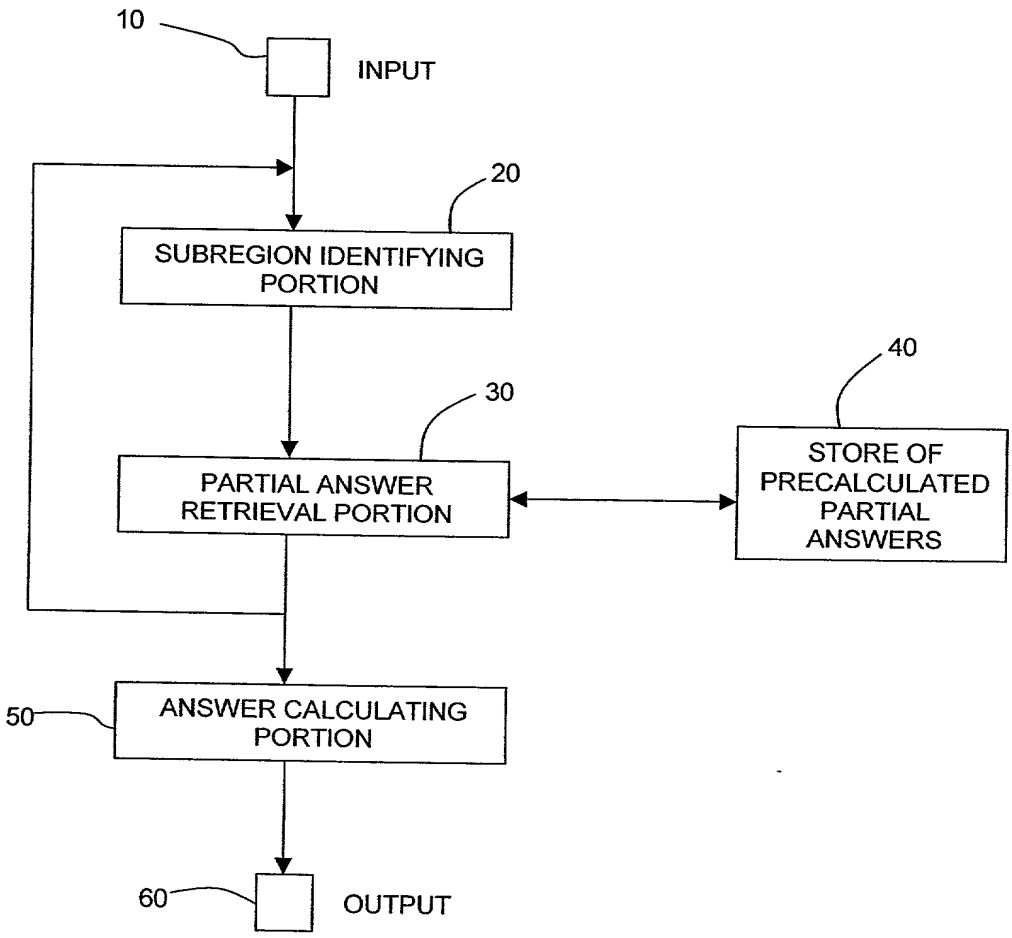

10
INPUT
20
SUBREGION IDENTIFYING PORTION
30
PARTIAL ANSWER RETRIEVAL PORTION
40
STORE OF PRECALCULATED PARTIAL ANSWERS
50
ANSWER CALCULATING PORTION
60
OUTPUT

FIG. 15

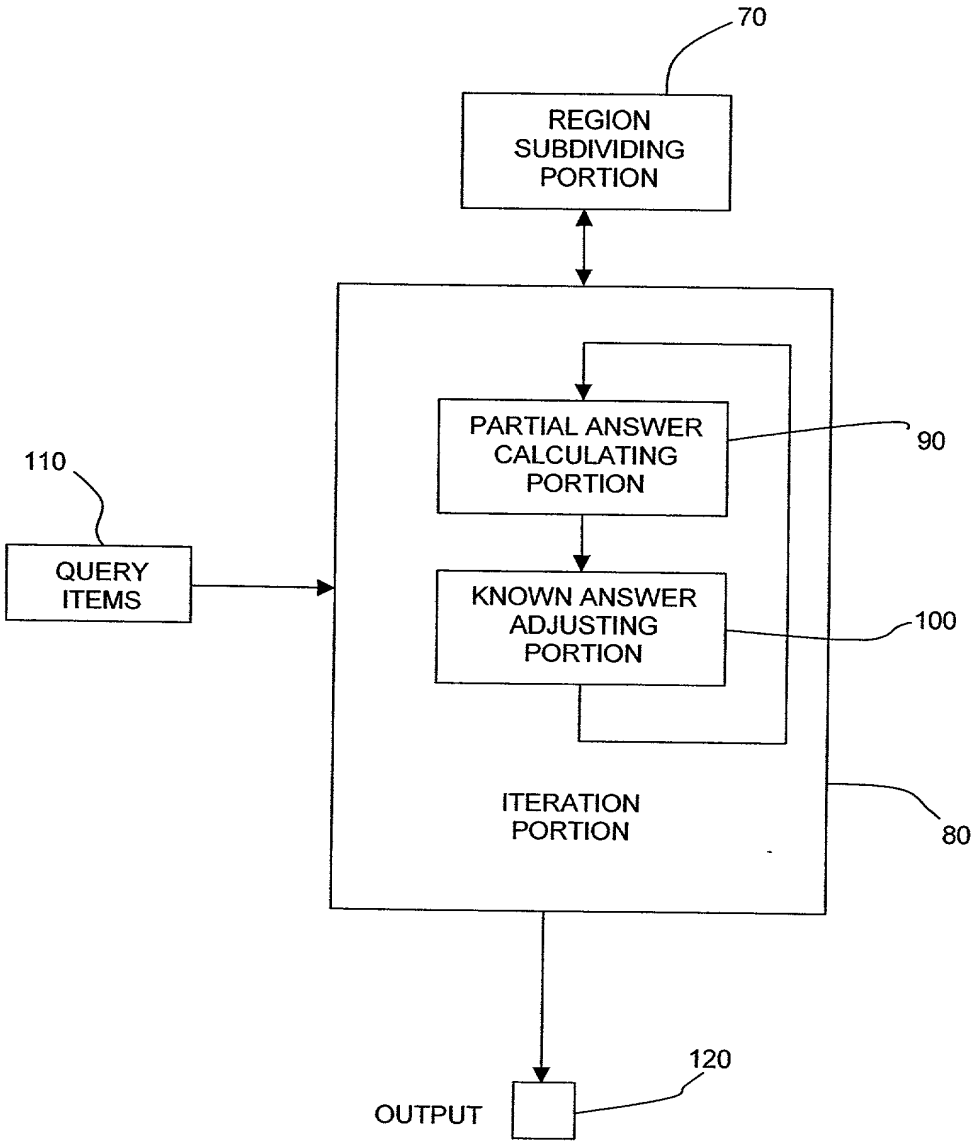

70
REGION SUBDIVIDING PORTION
PARTIAL ANSWER CALCULATING PORTION
90
110
QUERY ITEMS
KNOWN ANSWER ADJUSTING PORTION
100
ITERATION PORTION
80
120
OUTPUT

FIG. 16

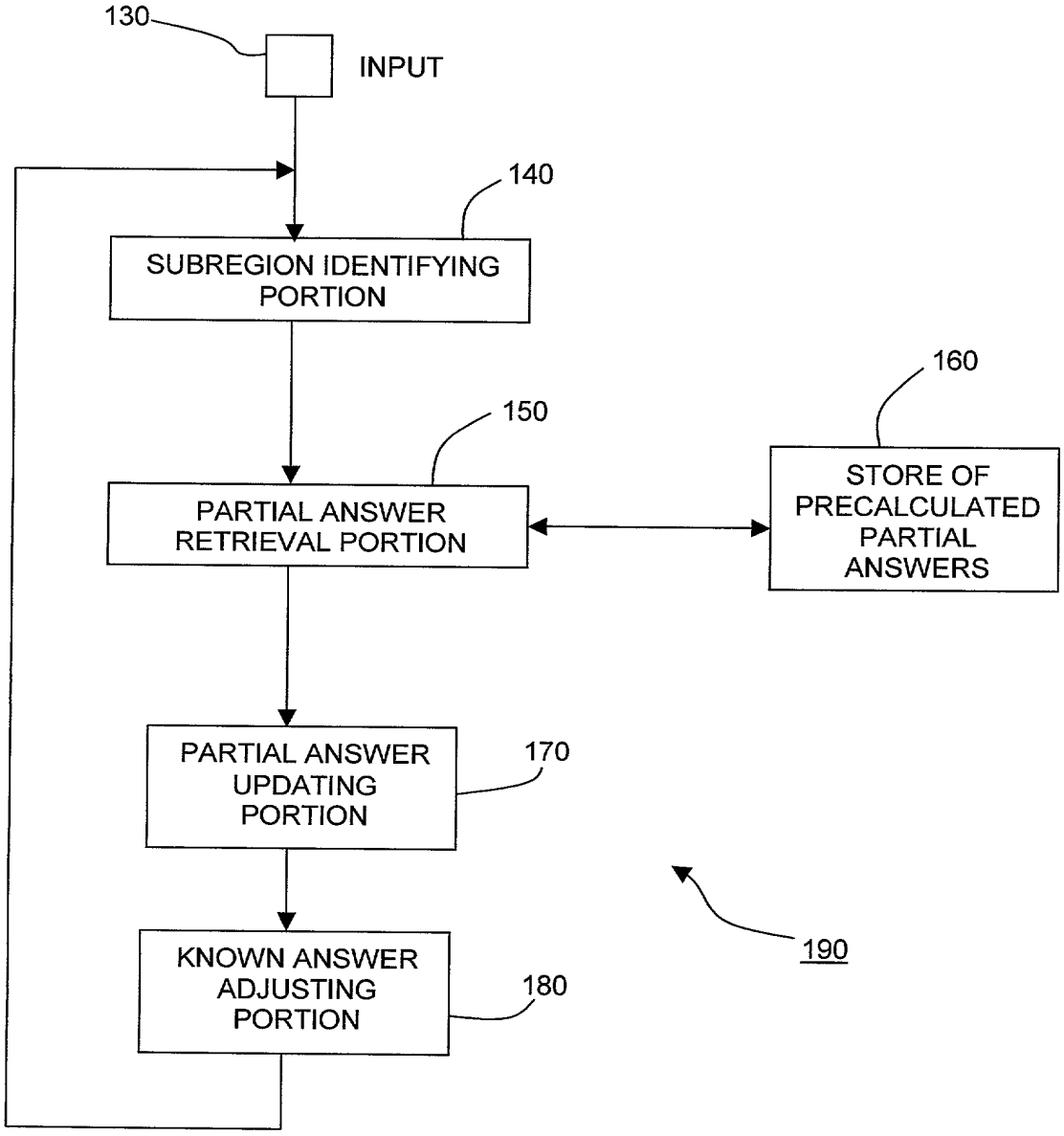

130
INPUT
140
SUBREGION IDENTIFYING PORTION
150
PARTIAL ANSWER RETRIEVAL PORTION
160
STORE OF PRECALCULATED PARTIAL ANSWERS
PARTIAL ANSWER UPDATING PORTION
170
KNOWN ANSWER ADJUSTING PORTION
180
190

FIG. 17

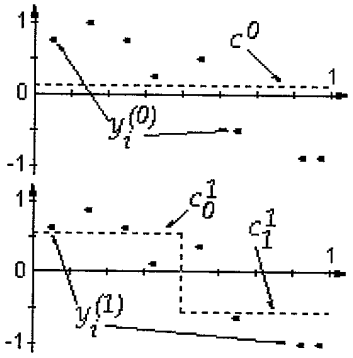


FIG. 18

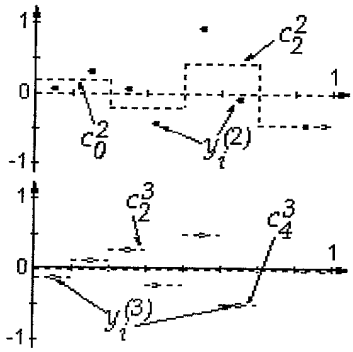


FIG. 19

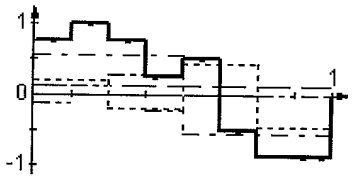


FIG. 20

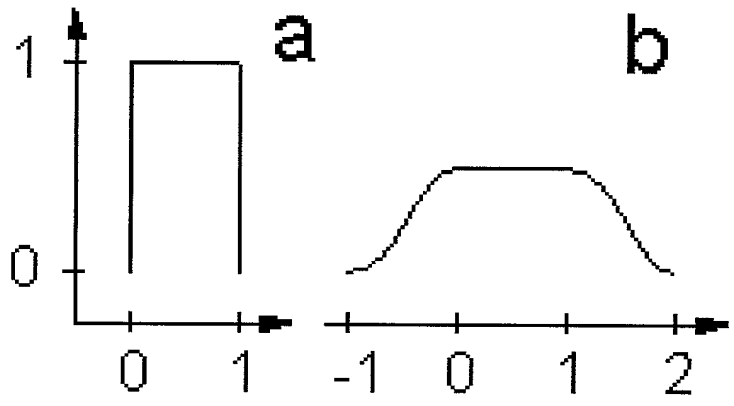


FIG. 21

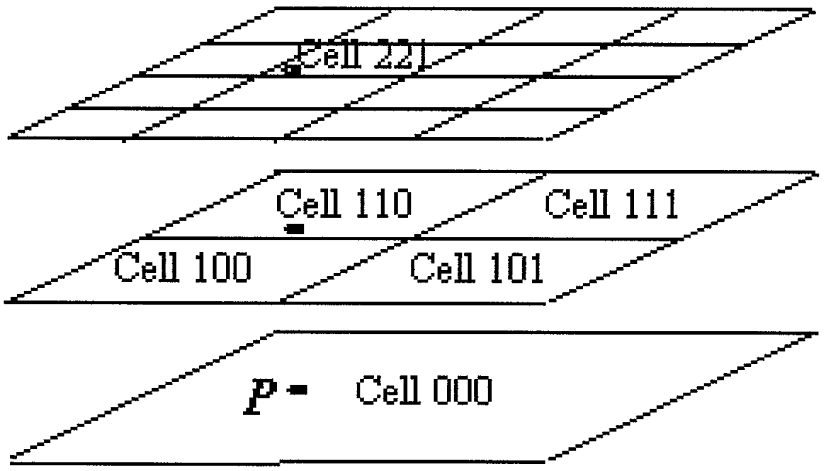

Cell 221
Cell 110
Cell 111
Cell 100
Cell 101
P
Cell 000

FIG. 22

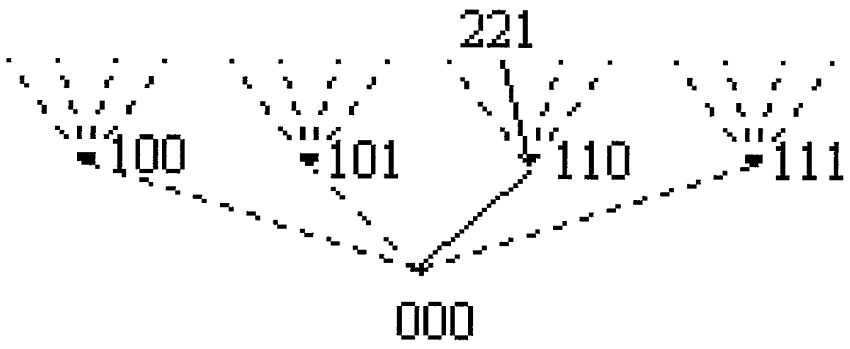

221
100
101
110
111
000

FIG. 23

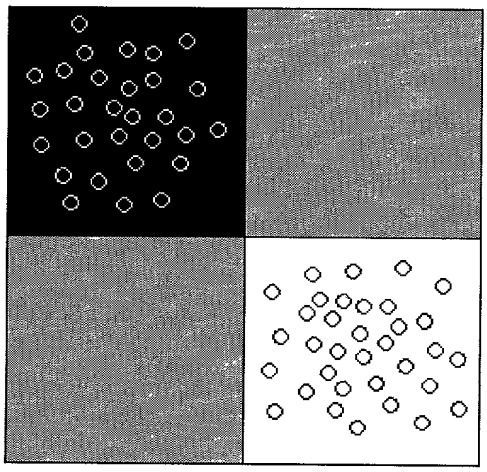
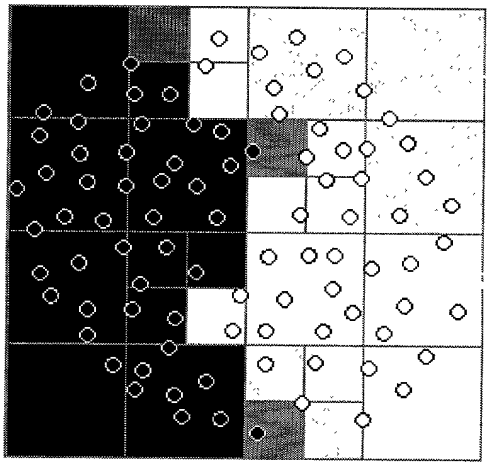


FIG. 24



20202D"595T60DT

FIG. 25

	Interpolation	Approximation
Training time (sec)	$24.8 \cdot 10^{-6}$	$24.8 \cdot 10^{-6}$
Query time (sec)	$11.62 \cdot 10^{-6}$	$16.35 \cdot 10^{-3}$
Memory required, (kbyte)	20	20
Train performance (%)	100.0	77.60
Query performance (%)	70.5	76.16

FIG. 26

Results of Garcke et al	# of points	training correctness	Testing correctness	runtime, sec		
Level 1	$5x10^4$	87.9	88.1	158		
	$5x10^5$	88.0	88.1	1570		
	$5x10^6$	88.0	88.0	15933		
Level 2	$5x10^4$	89.2	89.3	1155		
	$5x10^5$	89.4	89.2	11219		
	$5x10^6$	89.3	89.2	112656		
Results of This Work	# of points train& test	training correctness	Testing correctness	train time,sec	test time,sec	memory used
Level 1	50k & 50k	86.2	86.2	0.35	0.25	3.1k
	500k &500k	86.4	86.5	3.4	2.4	3.1k
Level 2	50k & 50k	91.6	89.3	0.75	0.45	163k
	500k &500k	90.7	90.5	9.5	4.5	197k
Level 3	50k & 50k	98.2	88.8	1.55	0.85	1.53M
	500k &500k	95.4	90.6	16.5	9.5	5.40M
Level 4	50k & 50k	99.9	88.8	2.05	1.15	3.86M
	500k &500k	99.5	90.3	21.5	12.5	24.7M
Level 5	50k & 50k	100.0	88.8	2.55	1.25	6.26M
	500k &500k	100.0	90.3	28.5	15.5	48.6M

Patent Application Publication